



ЕВРОПЕЙСКИ СЪЮЗ
ЕВРОПЕЙСКИ ФОНД ЗА
РЕГИОНАЛНО РАЗВИТИЕ



ОПЕРАТИВНА ПРОГРАМА
ИНОВАЦИИ И
КОНКУРЕНТОСПОСОБНОСТ

ДО
„Иннолоджика“ ООД
(Бенефициент- наименование)
гр. София, Студентски град, бул.
„Академик Борис Стефанов“ № 35,
Търговски център, ет.2, офис 30
(Адрес на бенефициента)

О Ф Е Р Т А

ОТ: _____
(наименование на кандидата)

за участие в процедура „Избор с публична покана“ за определяне на изпълнител с предмет:
**“Разработване, доставка и въвеждане в експлоатация на специализиран софтуер –
Специализирана софтуерна платформа за медиен мониторинг – 1 брой”**
(наименование на предмета на процедурата)

с адрес: гр. _____ ул. _____, № _____,
тел.: _____, факс: _____, e-mail: _____
регистриран по ф.д. № _____ / _____ г. по описа на _____ съд,
ЕИК /Булстат: _____,
представявано от _____, в качеството му на
_____.

УВАЖАЕМИ ГОСПОДА,

С настоящото Ви представяме нашата оферта за участие в обявената от Вас процедура за определяне на изпълнител с предмет:

**“Разработване, доставка и въвеждане в експлоатация на специализиран софтуер –
Специализирана софтуерна платформа за медиен мониторинг – 1 брой”**
(наименование на предмета на процедурата)

Декларираме, че сме разгледали документацията за участие и сме запознати с указанията и условията за участие в обявената от Вас процедура. Съгласни сме с поставените от Вас условия и ги приемаме без възражения.

Запознати сме и приемаме условията на проекта на договора. Ако бъдем определени за изпълнител, ще сключим договор в нормативноустановения срок.

Заявяваме, че при изпълнение на обекта на процедурата _____
подизпълнители. _____
ще ползваме/няма да ползваме

----- www.eufunds.bg -----

*Проект BG16RFOP002-1.001-0275-C01 „Внедряване на иновации в "Иннолоджика" ООД“, финансиран от
Оперативна програма „Иновации и конкурентоспособност“ 2014-2020, съфинансирана от Европейския съюз
чрез Европейския фонд за регионално развитие. Цялата отговорност за съдържанието на документа се носи
от "Иннолоджика" ООД и при никакви обстоятелства не може да се приема, че този документ отразява
официалното становище на Европейския съюз и Управляващия орган.*

Предлагаме срок за изпълнение на предмета на процедурата _____ календарни дни/месеца, считано от датата на подписване на договора за изпълнение.

Минималният срок за доставка и въвеждане в експлоатация, който участниците в процедурата могат да оферират е 2 месеца.

Декларираме, че представената от нас оферта е валидна до _____ (посочва се срокът, определен от бенефициента в публичната покана).

ТЕХНИЧЕСКО ПРЕДЛОЖЕНИЕ

Относно изискванията и условията, свързани с изпълнението на предмета на настоящата процедура, ще изпълним следното:

<p align="center">Изисквания и условия на</p> <p align="center">_____</p> <p align="center"><i>(наименование на бенефициента)</i></p>	<p align="center">Предложение на кандидата</p> <p align="center"><i>Марка/модел/производител/тех-нически характеристики</i></p>	<p align="center">Забележка</p>
<p>Изисквания към изпълнението и качеството на стоките / услугите / строителството:</p> <p>Специализирана софтуерна платформа за медиен мониторинг – 1 брой, с еквивалентни или по-добри параметри от следните минимални технически и функционални характеристики:</p> <p align="center">ТЕХНИЧЕСКА СПЕЦИФИКАЦИЯ</p> <p>по процедура за избор на изпълнител „Избор с публична покана“ с предмет: „Разработване, доставка и въвеждане в експлоатация на специализиран софтуер – Специализирана софтуерна платформа за медиен мониторинг – 1 брой“,</p> <p>по договор за безвъзмездна помощ BG16RFOP002-1.001-0275-C01 по процедура BG16RFOP002-1.001 „Подкрепа за внедряване на иновации в предприятията“, Оперативна програма „Иновации и конкурентоспособност“ 2014-2020 г.”</p> <p><u>Специализирана софтуерна платформа за медиен мониторинг – 1 брой, с еквивалентни или по-добри параметри от следните минимални технически и функционални характеристики:</u></p>		

1. Обзор на системата и основно описание

Специализираната софтуерна платформа за медия мониторинг трябва да представлява комплексна система състояща се от бекенд част, способна да обновява милиони източници на новини (емисии) в рамките на минути, да сверява старото с новото съдържание на емисиите и да записва само новото съдържание в база данни. Всяка записана статия трябва да минава през редица филтри и правила, които всеки от потребителите на системата може да заложи и на база на които трябва да се предприемат действия – известяване, категоризиране в отчети и други описани по-подробно по-надолу в документа. Архитектурата на системата и базата данни трябва да е съобразена с обема, който ще постъпва. Системата трябва да е калкулирана да обработва и съхранява до 20,000,000 (20 милиона) статии всеки ден, да може да изтегля достатъчно бързо и паралелно емисиите, така че да може да обработва до 1 милион източника на час. За целта възлите на системата трябва да са оптимално добре издържани архитектурно и кодът да е написан на възможно най-ниското ниво, което позволява програмният език.

Потребителите трябва да могат да достъпват системата чрез Web интерфейс и мобилни приложения за Android, iOS и Windows Phone.

Напредналите потребители и разработчици трябва да могат да достъпват системата и чрез приложно програмен интерфейс (API).

Системата трябва да има добър презентационен сайт (landing page) където да е показано какво прави, какви нужди удовлетворява и какви проблеми решава.

Тъй като ще се таргетира международна аудитория, системата трябва да бъде с основен език Английски и с вградена възможност за преводи. Да бъде предоставена система за автоматично превеждане от потребителите, за да бъде постигнат ефекта на crowdsourcing.

Основни характеристики:

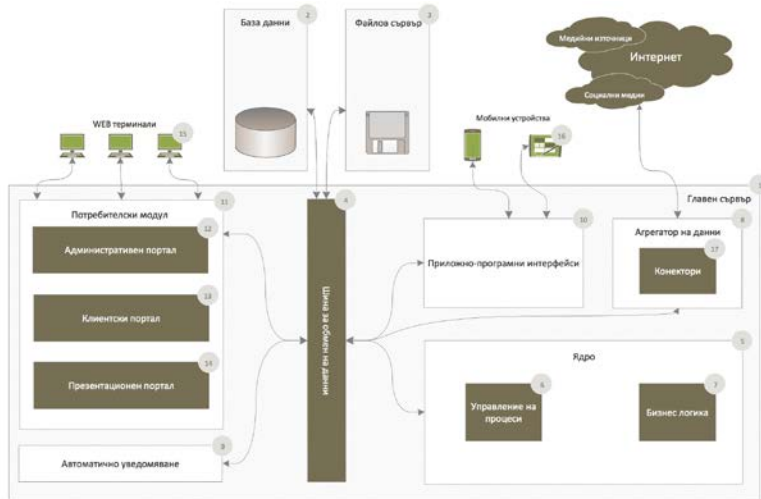
- Възможност за следене на източници чрез RSS протокол;
- Възможност за следене на източници чрез OAuth2 оторизация и REST интерфейси;
- Възможност за следене на източници чрез входящи email съобщения;
- Разработване на модел за база данни, способен да съхранява неограничено количество информация чрез прилагане на добри практики за скалируемост и big-data;
- Разработване на приложни интерфейси за надграждане на платформата;
- Разработване на механизъм за автентикация на потребители;
- Разработване на абстрактен аналитичен модул за анализ на съдържанието на новините и съвпадението му с едно или

<p>повече предварително въведени правила.</p> <p>Нефункционални спецификации:</p> <ul style="list-style-type: none"> – Програмен език: PHP; – Бази данни: MySQL; – Системата да бъде оптимизирана за работа върху POSIX операционни системи (Linux/UNIX); – За автентикация на потребителите да се използват само съвременни методи за криптиране (bcrypt) на пароли. <p>2. Архитектура на Специализирана софтуерна платформа за медиен мониторинг</p> <p>Архитектурна стратегия</p> <p>Системата трябва да следва LAMP принципите, поради следните причини:</p> <ul style="list-style-type: none"> • Прозрачност и отворена среда на компонентите • Огромна база от потребители и разработчици • Безплатни дистрибуции и компоненти • Бърза разработка и поддръжка на програмните езици и компоненти <p>Системата от гледна точка на потребителите трябва да бъде облачна услуга. При използване на Web browser потребителите не трябва да инсталират нищо, а само да насочат браузъра си към уеб сайта на системата. При използване на мобилно устройство с Android, iOS или Windows Phone, потребителите инсталират приложение от съответният App Store, което работи като тънък клиент – използва приложно-програмният интерфейс на системата за оторизация, извличане и промяна на данни.</p> <p>Задължително е системата да се инсталира на хардуер на „Иннолоджика“ ООД.</p> <p>Сървърна операционна система – Linux. Задължително поддръжжани версии – CentOS 6 и нагоре, RedHat 6 и нагоре.</p> <p>Уеб сървъри – Apache/mod_php, Apache/php-fpm или NGINX/php-fpm. Трябва еднакво добре да се поддържат и двата уеб сървъра и варианти на PHP.</p> <p>Програмен език сървърна част – PHP. Задължителна поддръжка на PHP 7</p> <p>Програмен език клиентска част (Web) – JavaScript, CSS, HTML.</p> <p>Програмен език клиентска част (iOS) - Visual C</p> <p>Програмен език клиентска част (Android) - Java</p> <p>Програмен език клиентска част (Windows Phone) - C#</p> <p>Бази данни – MySQL, Elasticsearch. Задължително е</p>		
--	--	--

използването само на MySQL 5.5 и нагоре, както и Percona Server. Elasticsearch не трябва да използва като основна, а само помощна база за търсене.

Позволените библиотеки - [Yii2](#), jQuery, Vuejs, Bootstrap

Желана архитектура



Крайната система трябва да се придържа максимално към желана архитектура изобразена по-горе. Задължителен елемент е шината за обмен на данни (4). Тя е абстрактен слой чрез който различните модули обменят информация помежду си. Комуникация между модулите по други канали не бива да се осъществява. В шината трябва да може да се залагат различни прихващания (hooks) чрез които системата да може да се надгражда допълнително чрез плъгини.

Системата за автоматично уведомяване (9) трябва да поддържа уведомяване чрез следните методи:

- Email
- SMS
- Push notifications (Android и iOS)

Ядрото (5) трябва да съдържа в себе си модул за управление на процеси (6) и цялата бизнес логика в отделна абстракция, която да може да се разширява при нужда. Модулът за управление на процеси определя начинът на стартиране на backend процесите. Трябва да може да се стартират процеси по три начина:

- През интервал – процесът се стартира, извършва дейността си и излиза. Модулът следи за изтичане на зададения интервал и го стартира отново. Така цикълът се затваря
- Периодично – задават се точни периоди на които процесът се стартира – примерно на всеки 5 минути, всеки час. Всеки понеделник в 10:00 и т.н. Преди стартиране на процеса се прави проверка дали той вече не работи. Ако работи не се

стартира повторно за да се предотврати проблем с данните

- Персистентен процес (watchdog) – в този режим се пускат процеси, които никога не трябва да спират. Ако случайно някой процес спре (поради фатална грешка например или друга непредвидена причина) той моментално трябва да се стартира отново.

В бизнес логиката (7) трябва да е заложена цялата логика по обработка на получените статии от източниците. Тя трябва да е добре структурирана, така че да може да се доразвива.

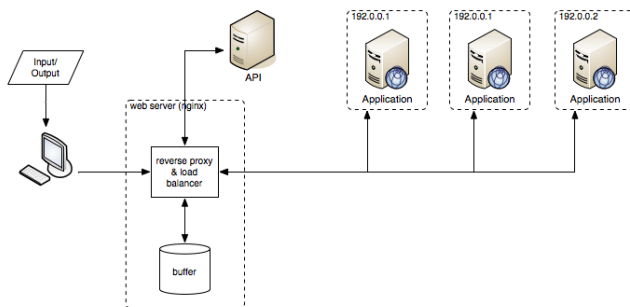
Скалируемост

Специализираната софтуерна платформа за медиен мониторинг трябва да може да работи при следните параметри:

- До 10,000 едновременно логнати потребители на уеб интерфейса
- До 5,000 едновременно логнати потребители през мобилни приложения
- Да може да издържа до 20,000,000 нови статии на ден.
- Да може да складира статиите до безкрай само чрез добавяне на нови сървъри.
- Да може да обновява до 2 милиона източника на емисии в рамките на един час.

Надежност

Специализираната софтуерна платформа за медиен мониторинг трябва да има предвидени вградени механизми за надежност и непрекъсваемост на услугата. Задължително е уеб сървърите да могат да се скалират хоризонтално при нужда. За целта трябва да се използва комбинация от NGINX и Apache сървъри както е на показаната архитектура:



Диаграма на уеб архитектурата на уеб сървърите

За надежност на базите данни трябва да се използва

асинхронната репликация на MySQL, като превключването към Slave база не трябва да бъде автоматизирано.

Преизползване

Специализираната софтуерна платформа за медиен мониторинг трябва да бъде разработена при спазване на най-добрите практики в програмирането. Съвърнатата част трябва да бъде изцяло обектно ориентирана, като всички обекти, методи и свойства трябва да бъдат добре структурирани, така че да позволяват лесно преизползване в случай на нужда на разширяване на системата.

Производителност

Специализираната софтуерна платформа за медиен мониторинг трябва да бъде достатъчно оптимизирана, за да може да работи при параметрите описани в т. **Скалируемост**, без забележимо забавяне към потребителят. Средното време за отговор на приложно-програмният интерфейс не трябва да бъде по-високо от 20ms. Средното време за извеждане на страница в Web интерфейса не трябва да бъде повече от 200ms.

Оперативна съвместимост

Специализираната софтуерна платформа за медиен мониторинг трябва да може да работи както автономно, така и да получава данни от външен източник на информация чрез консумиране на приложно-програмен интерфейс.

Портативност

Специализираната софтуерна платформа за медиен мониторингне трябва да бъде обвързана със специфичен хардуер и операционна система. Кода на системата трябва да се съхранява в git хранилище и при нужда да може да бъде изтеглен на нов сървър.

Сигурност

Специализираната софтуерна платформа за медиен мониторинг НЕ трябва да съдържа нито една от долупосочените уязвимости:

- Code injection: хакерите да намерят начини да вмъкват зловреден изпълним код чрез легитимен трафик.
- Слаба автентификация и управление на сесиите: компрометираща самоличността на потребителите в най-различни начини
- Cross-сайт скриптове: подобно на инжектирането на код, но със скриптове изтеглени от несигурни източници
- Несигурните препратки към файлове: получаване на достъп до файлове, до които уеб сървърът не

<p>трябва да има достъп</p> <ul style="list-style-type: none"> • <i>Излагане на чувствителни данни: Недобра защита на личните данни на потребителите (Имена, email адреси, телефонни номера), което в следствие да бъде използвано от недоброжелателни лица.</i> • <i>Слаб контрол на нивата на достъп: Слаба проверка на текущото ниво на достъп на потребителят, даваща възможност на недоброжелател да повдигне нивото си на достъп по изкуствен път.</i> • <i>Cross-site request forgery: Компрометиране на приложението, чрез предоставяне на валидирана, но подправена информация от трета страна.</i> • <i>Компоненти с известни уязвимости: Използване на компоненти, които имат публичен достъп и имат известни уязвимости. Минимизирането на външни компоненти намалява този риск.</i> <p>Всяка комуникация през потребителските и приложно-програмните интерфейси трябва да се осъществява посредством Secure Sockets Layer (SSL) без изключение.</p> <p>Обработка на грешки</p> <p><u>Специализираната софтуерна платформа за медиен мониторинг</u> трябва да поддържа различни нива на записване на грешки, които да могат да бъдат преглеждани по всяко едно време:</p> <ul style="list-style-type: none"> • <i>Всеки unhandled exceptions PHP трябва да се записва.</i> • <i>Всяка MySQL грешка трябва да се записва заедно с дата и време, IP адрес на сървъра от който е дошла, както и потребител в интерфейса.</i> • <i>В процеса на събиране на данни от източниците трябва да се записва всяка грешка при изтегляне или анализиране на XML файловете.</i> • <i>Логовете трябва периодично да се редуват и архивират.</i> <p>Мониторинг и защита на процесите</p> <p>Всички backend процеси трябва да се мониторират:</p> <ul style="list-style-type: none"> • <i>Един и същи процес да не се стартира два или повече пъти в паралел.</i> • <i>Дали един процес не работи подозрително дълго време (времето трябва да се конфигурира поотделно за всеки от процесите). Ако има такива системата</i> 		
--	--	--

трябва да алармира администраторите по email.

- Процесите трябва да изпращат т.нар. heartbeats към централизирано място където да се проверява периодически, че всеки процес изпраща регулярно такива. Ако някой процес не изпраща системата трябва да алармира администраторите по email.

Модул “Ядро”

- Съдържа основната бизнес логика на приложението;
- Съдържа подсистема за управление на процеси и задачи. Дава възможност да се определят интервалите на които да се стартират определените задачи, какво да бъде максималното им работно време и запазва изхода от тяхната работа;
- Достъпа и ограничението на потребителите се осъществява в този модул;
- Паролите на потребителите са криптирани с mcgurt модул;
- Поддържат се следните външни доставчици на оторизация: Google, Facebook, Twitter, LinkedIn;

Модул “Потребителски”

- Представява целият потребителски интерфейс на приложението;
- Да бъде написан използвайки модерни технологии и похвати – HTML5, jQuery, Bootstrap;
- Включва страница за презентация на системата (landing page);
- Поддържа нива на достъп, като има отделни потребителски и административни интерфейси;

Модулът трябва да е т.нар. уеб фронтенд на приложението. Модулът трябва да е разделен на три основни части (портали) – презентационен портал, административен портал и потребителски (клиентски) портал. Те са разделени функционално, спрямо различните цели и нужди на потребителите и на „Иннолоджика“ ООД.

Презентационен портал

Целта на презентационния портал е да е презентира функционалността на системата (Специализираната софтуерна платформа за медиен мониторинг) пред нови и потенциални потребители. Представява съвкупност от лендинг страници, на които попадат потребителите, които не са регистрирани и вписани в системата.

Основният език на презентационния портал трябва да бъде Английски, като трябва да има опция за смяна на езика. Трябва да се поддържа и Български, както и да бъде предложена схема за превеждане от потребители.

Презентационният портал трябва ясно и точно да изтъква предимствата на системата, да показва как работи тя и да дава бърза възможност за регистрация с ясно изразени call-to-action бутони. Трябва също така да дава информация за цените и плановете на приложението. Задължително е да присъстват политика на конфиденциалност и условия за ползване.

На презентационният портал трябва да бъдат поставени на видно място логата на ЕС и Оперативна програма Иновации и конкурентоспособност 2014-2020. Файловете могат да бъдат намерени в оригинал на адрес <http://opic.bg/informatsiya-i-publichnost/grafichni-iziskvaniya>

Примери за добре разработени презентационни портали са:

Mention (<https://mention.com>):

Brand24 (<https://brand24.com/>):

Трябва да се поддържат основните методи за регистрация и вписване, които са Email/парола или социални мрежи (Facebook, Google, LinkedIn). Тези методи трябва да бъдат разработени в приложението и бекенда.

- Facebook login - <https://developers.facebook.com/docs/facebook-login/web>
- Google login - <https://developers.google.com/identity/sign-in/web/>
- LinkedIn login - <https://developer.linkedin.com/docs/oauth2>

Потребителски портал

Потребителският портал е основната част, с която се борава в системата.

Основният език трябва да бъде Английски, като трябва да има опция за смяна на езика. Трябва да се поддържа и Български език, както и да бъде предложена схема за превеждане от потребители.

Потребителският портал трябва да съдържа следните секции:

- Настройки
- Репорти
- Ключовидуми
- Статии
- Табло (Dashboard)

Настройки

Потребителят трябва да има право да променя следните настройки:

- *Име*
- *Потребителско име*
- *Email*
- *Парола*
- *Език*
- *Честота на операсняване на новините*
- *Включване/изключване на известия за новини (чрез [Web Notifications](#))*

Репорти

Потребителите трябва да могат да генерират неограничено количество репорти.

Репортите са основно списъци от статии, които са съвпаднали с дадени ключови думи. В един репорт може да участват една или повече ключови думи. Репортите трябва да могат да се разглеждат по дни, седмици и месеци. Освен списъци от статии и съвпадения в репортите трябва да има още:

- *Графика на съвпаденията по дни*
- *Гъстота на ключовите думи (коя дума се среща най-често в репорта)*
- *Езици на статиите*

Ключови думи и правила

Ключовите думи реално са правила, които потребителите дефинират и на база на които правила определени статии съвпадащи с тях трябва да се добавят към репорти или да се изпращат към модула за автоматично известяване. Следните правила трябва да може да се дефинират от потребителите:

- *Ключова дума присъства в заглавие/съдържание/автор*
- *Ключова дума не присъства в заглавие/съдържание/автор*
- *Ключова дума е в началото на заглавие/съдържание/автор*
- *Ключова дума е в края на заглавие/съдържание/автор*

Освен по ключови думи, потребителите трябва да могат да дефинират и регулярни изрази в търсенето си. Например

/бойко(s+)борисов/i за търсене на "Бойко Борисов" без значение от броят интервали, нови редове и табулации между двете думи.

Едно правило в потребителския интерфейс трябва да може да съдържа комбинация от различни ключови думи и типове на съвпадения.

Едно правило трябва да има избор дали да смята съвпадение при която и да е ключова дума или при съвпадение на всички.

Статии

Основна функционалност на системата е извличането на статии от медийни източници. Тези статии трябва да се визуализират подобаващо в потребителския интерфейс, като трябва да има различни режими на визуализация – списък, списание, картички. В режим списък, статиите трябва да се визуализират само по заглавия, като при избиране на статия тя се отваря в пълният ѝ вид. В режим списание, освен заглавието в списъка трябва да се появява и снимка от статията, както и първите няколко реда от съдържанието. В режим картички трябва да се виждат големи снимки от статиите, както и заглавието на статията. В първите два режима статиите трябва да са по една на ред, а в режим картички да се виждат повече статии в зависимост от големината на екрана.

Табло (Dashboard)

В таблото потребителите трябва да виждат основната информация, която може да ги интересува след влизане в системата. Това трябва и да бъде първият екран, който се визуализира след вход. Там трябва да се виждат следните неща:

- *Новини от системата (обновления)*
- *Линкове към мобилните устройства*
- *Брой нови статии от последното влизане*
- *Генерирани нови репорти*
- *Списък с ключовите думи и брояч след всяка дума показващ броя съвпадения за дена*
- *Непрочетени нотификации*
- *Графика на статиите в репорти по дни*
- *Графика на гъстотата на ключовите думи*

Административен портал

Трябва да може да се използва от служители на „Иннолоджика“ ООД, които да подреждат новинарският еемисии, да преглеждат

<p>потребителите и техните плащания, да могат да ги обслужват пълноценно; да виждат различни статуси от модули на системата, например опашки от емисии за изтегляне, брой логнати потребители, брой регистрирани потребители за последният час/ден/седмица/месец.</p> <p><u>Шина за обмен на данни (ESB):</u></p> <ul style="list-style-type: none"> - С цел по-добра модулност на системата, да се използва обща шина за обмен на данни между модулите; - Комуникация между модулите чрез JSON; - Възможност за логика в самата шина при постъпване на определени събития; <p><u>Модул “Агрегатор на данни”</u></p> <ul style="list-style-type: none"> - Сървърно независим - При нужда да може да се добавят допълнителни сървъри за хоризонтално скалиране; - Работа с много паралелни процеси за постигане на оптимална работа при изтегляне на огромен брой източници едновременно; - Може да чете източници по RSS, JSON, XML; - Поддържа Basic HTTP Authentication; - Поддържа OAuth2; - Използва XML Tidy за поправяне на проблемни RSS източници, така че да може да се прочитат коректно максимален брой; - Разполага с конфигурация за броя процеси, които да се използват при изтегляне; <p><u>Модул “Агрегатор на данни”</u> трябва да представлява система с висока производителност, способна да събира своевременно данни от различни източници, да ги уеднаквява до обекти от тип статия и да ги съхранява в базата данни за по-нататъшна обработка и презентация. <u>Модул “Агрегатор на данни”</u> трябва да може да скалира хоризонтално при нужда чрез добавяне на нови сървъри, които да обработват опашките от емисии.</p> <p><u>Модул “Агрегатор на данни”</u> трябва да се състои от няколко части:</p> <p>Транспортна част – отговаря за изтеглянето на емисиите. Трябва да има сълно паралелизирана функционалност, устойчивост на мрежови грешки и опашка с повторни опити (exponential backoff). Тази част не трябва да съдържа бизнес логика и анализ на получените данни. Задължителна е поддръжката на HTTP Conditional GET за оптимизиране на мрежовият трафик и освобождаване на анализиращата част от излишна дейност. Примерен HTTP Conditional GET поток:</p>		
--	--	--

```
curl "http://blog.jqueryui.com/feed/" --header 'If-Modified-Since:
Wed, 01 Mar 2017 20:24:24 GMT' --header 'If-None-Match:
"cfed101f517d6b038f6f7129acecb647"' --silent -head
HTTP/1.1 304 Not Modified
Date: Wed, 01 Mar 2017 07:32:28 GMT
Connection: keep-alive
Set-Cookie: __cfduid=d234947179b9fd99e6535cc442fb617f81488353547;
expires=Thu, 01-Mar-18 07:32:27 GMT; path=/; domain=.jqueryui.com;
HttpOnly
Last-Modified: Thu, 15 Sep 2016 16:51:39 GMT
ETag: "cfed101f517d6b038f6f7129acecb647"
Link: <http://blog.jqueryui.com/wp-json/>; rel="https://api.w.org/"
Server: cloudflare-nginx
CF-RAY: 338a6c29334264d5-FRA
```

Отсрещният сървър връща код 304 без съдържание в body-то, което означава, че документът не се е променял от предишното изтегляне.

Транспортната част също така трябва да може да борави със всички OAuth токени на потребителите при изтегляне на емисии от REST интерфейси изискващи оторизация. Това са:

- Twitter REST API - <https://dev.twitter.com/rest/public>
- Facebook Graph API - <https://developers.facebook.com/docs/graph-api>
- Google+ API - <https://developers.google.com/+/web/api/rest/>
- VK API - <https://vk.com/dev/openapi>

В случай на грешка при изтегляне, това трябва да се отбелязва в лог файл на сървъра както и към самата емисия в базата данни. Последното е важно за да може потребителите да разбират кога дадена емисия изпитва затруднения и не се получават новини от нея.

Анализираща част (Parser) – отговаря за обработка на получените данни от транспортната част. Изходът от анализиращата част трябва да бъде унифициран обект, тип статия със следните свойства:

- Уникален идентификатор
- Заглавие
- Автор
- Съдържание (summary)
- Линк
- Дата на получаване (с точност до **микросекунда**)
- Дата на публикуване (с точност до секунда)
- Дата на последно обновяване (с точност до секунда)
- Списък с категории ([link](#))
- Език (от детекторът на езици)

Форматите, които трябва да се поддържат от

www.eufunds.bg

14

<p><i>анализиращата част без допълнителна разработка са:</i></p> <ul style="list-style-type: none"> • <i>RSS 0.90</i> • <i>RSS 0.91 (Netscape)</i> • <i>RSS 0.91 (Userland)</i> • <i>RSS 0.92</i> • <i>RSS 1.0</i> • <i>RSS 2.0</i> • <i>Atom 0.3</i> • <i>Atom 1.0</i> • <i>JSON</i> • <i>Pure XML</i> <p><i>Допълнителните именни пространства (link), които трябва да се поддържат са:</i></p> <ul style="list-style-type: none"> • <i>Dublin Core 1.0</i> • <i>Dublin Core 1.1</i> • <i>GeoRSS</i> • <i>iTunes RSS 1.0 (mostly complete)</i> • <i>Media RSS 1.1.1</i> • <i>RSS 1.0 Content Module</i> • <i>W3C WGS84 Basic Geo</i> • <i>XML 1.0</i> • <i>XHTML 1.0</i> <p><i>В допълнение за всеки от поддържаните JSON интерфейси трябва да има разширяема логика, чрез която избрани полета да се свеждат до свойства на статията.</i></p> <p><i>Уникалният идентификатор на статията трябва да се определя в зависимост от вида на емисията и наличността на полетата. Той трябва еднозначно да идентифицира дадената статия за да се предотврати повторното и постъпване в базата данни при последващо изтегляне на същата емисия.</i></p> <p><i>Датата на получаване трябва да бъде в микросекунди, тъй като се очаква в рамките на една секунда да се получават множество статии.</i></p> <p><i>Свойствата Автор и Заглавие трябва да бъдат почистени от HTML и друг код.</i></p> <p><i>Свойството съдържание може да съдържа HTML, но трябва да бъде почистено, така че да не съдържа злобен код и елементи, които биха компрометирали поведението на потребителския интерфейс в последствие. Единствено следните HTML елементи са допустими:</i></p>		
---	--	--

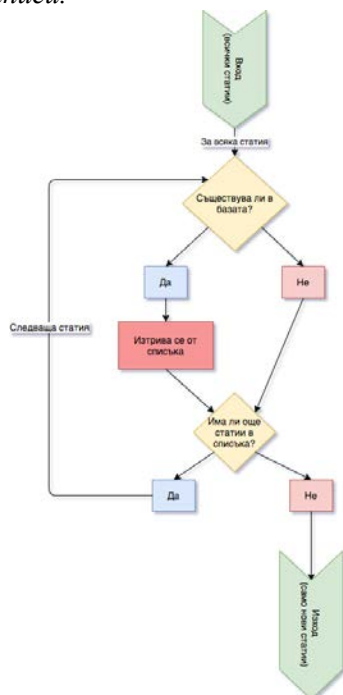
<p>a, em, i, p, br, b, strong, iframe, img, table, th, tr, td, div, span, pre, code, blockquote, h1, h2, h3, h4, h5, h6, small, ul, ol, li, tt, big, small, hr, address, cite, fdn, abbr, q, sub, sup, ins, del, caption, tfoot, thead, tbody, colgroup, col, object, embed, param, video, audio, source, strike</p>		
<p><i>Единствено следните атрибути на HTML елементи са допустими:</i></p>		
<p>src, style, href, title, alt, width, height, colspan, rowspan, dir, lang, frameborder, allowfullscreen, webkitallowfullscreen, mozallowfullscreen, name, value, type, preload, poster, loop, muted, autoplay, controls</p>		
<p><i>Всички препратки с релативен адрес трябва да бъдат пренаписани към абсолютен използвайки домейнът на емисията. Детекторът на език е компонент, който трябва с голяма точност да определи език на който е написана статията. Езиците, които трябва да бъдат поддържани от детектора са следните:</i></p>		
<ul style="list-style-type: none"> • <i>Afrikaans</i> • <i>Arabic</i> • <i>Bosnian</i> • <i>Bulgarian</i> • <i>Catalan</i> • <i>Chinese Simplified</i> • <i>Chinese Traditional</i> • <i>Croatian</i> • <i>Czech</i> • <i>Danish</i> • <i>Dutch</i> • <i>English</i> • <i>Estonian</i> • <i>Finnish</i> • <i>French</i> • <i>German</i> • <i>Greek</i> • <i>Hebrew</i> • <i>Hindi</i> • <i>Hungarian</i> • <i>Indonesian</i> • <i>Italian</i> • <i>Japanese</i> 		

- *Korean*
- *Latvian*
- *Lithuanian*
- *Malay*
- *Maltese*
- *Norwegian*
- *Persian*
- *Polish*
- *Portuguese*
- *Romanian*
- *Russian*
- *Serbian*
- *Slovak*
- *Slovenian*
- *Spanish*
- *Swedish*
- *Thai*
- *Turkish*
- *Ukrainian*
- *Urdu*
- *Vietnamese*
- *Welsh*

В случай на грешка при анализиране, това трябва да се отбелязва в лог файл на сървъра както и към самата емисия в базата данни, за да може потребителите да разбират кога дадена емисия изпитва затруднения и не се получават новини от нея.

***Дедупликираща част** – при успешно анализиране, списъкът със статии трябва да се подаде към дедупликиращата част. Целият модул **Агрегатор на данни** трябва да работи на принципа “изтегляне на всичко, добавяне само на новото”. Статии, които вече присъстват в базата данни трябва да бъдат спрени от дедупликиращата част. Изхода от тази част трябва да бъде списък с нови статии. Изходът може да бъде и празен списък, което прекратява процеса по обновяване на текущата емисия, тъй като се смята че няма нови статии. Проверката за съществуване на стари статии трябва да става по оптимален начин, тъй като се очаква да се извършват хиляди подобни проверки в секунда. Нужен е кеширащ механизъм от ниско ниво в*

паметта, който да разтовари базата данни от постоянни проверки в таблици с милиарди записи.



Диаграма на работата на дедупликираният модул

Проверка на ключови думи – в тази част всяка от статиите преминали през дедупликиращата част трябва да преминат проверка за съответствие по ключови думи, които потребителите на системата трябва да могат да дефинират. Следните правила трябва да може да се дефинират от потребителите:

- Ключова дума присъства в заглавие/съдържание/автор
- Ключова дума не присъства в заглавие/съдържание/автор
- Ключова дума е в началото на заглавие/съдържание/автор
- Ключова дума е в края на заглавие/съдържание/автор

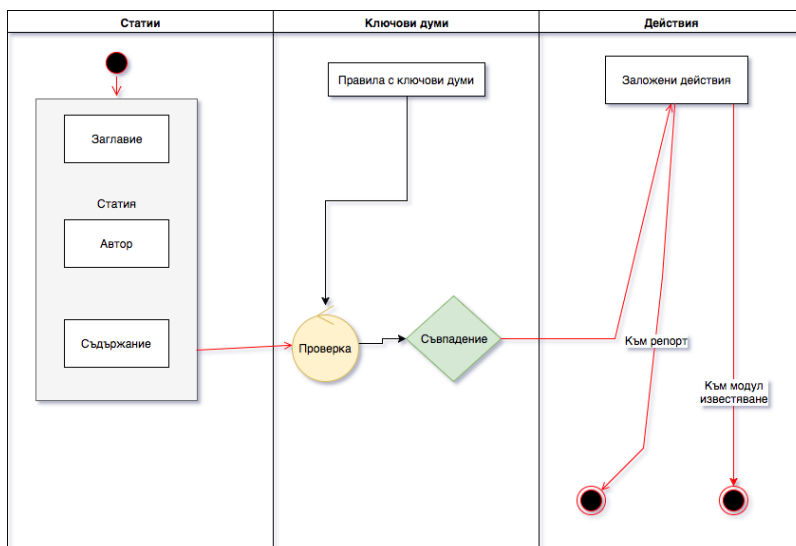
Освен по ключови думи, потребителите трябва да могат да дефинират и регулярни изрази в търсенето си. Например /бойко(\s+)борисов/i за търсене на "Бойко Борисов" без значение от броят интервали, нови редове и табулации между двете думи.

Едно правило в потребителския интерфейс трябва да може да съдържа комбинация от различни ключови думи и типове на съвпадения.

Едно правило трябва да има избор дали да смята съвпадение при която и да е ключова дума или при съвпадение на всички.

При изпълнение на правило (има съвпадения) трябва да могат да се предприема комбинация от следните действия:

- Добавяне на статията към репорт
- Изпращане на статията по email
- Изпращане на SMS нотификация
- Изпращане на Push нотификация към мобилни устройства



Проверка на ключови думи и предприемане на действия

Модул “Приложно-програмни интерфейси”

- Превежда вътрешните интерфейси от ниско ниво в JSON интерфейси за използване на външни приложения;
- Поддържа OAuth2 и Username/Password authentication;
- Поддържа rate-limiting за намаляване на риска от претоварване на системата от външни системи;

Приложно-програмните интерфейси, трябва да използват REST (Representational state transfer) - разпределителна системна рамка, базирана на уеб протоколи и технологии. Архитектурният модел REST включва взаимодействията между сървър и клиент, осъществени по време на трансфера на данни.

Основният ендпойнт трябва да бъде достъпен на адрес <https://system/api/0/>.

Отговорите на методите трябва винаги да съдържат коректен HTTP response код.

Таблица с кодовете:

Код	Описание
200	Заявката и отговорът са коректни

400	Липсващ задължителен входящ параметър
401	Крайният потребител не е оторизиран
404	Методър не е имплементиран
503	Услугата не е налична – например проблем с база данни

Примерна комуникация с грешка:

```
> POST /accounts/ClientLogin HTTP/1.1
> User-Agent: curl/7.19.7 (x86_64-redhat-linux-gnu) libcurl/7.19.7
NSS/3.14.0.0 zlib/1.2.3 libidn/1.18 libssh2/1.4.2
> Host: media-monitoring.com
> Accept: */*
> Content-Length: 27
> Content-Type: application/x-www-form-urlencoded
>
< HTTP/1.1 503 Service Temporarily Unavailable
< Date: Wed, 03 Feb2017 11:01:32 GMT
< Server: Apache
< X-Powered-By: PHP/7.1
< Vary: Accept-Encoding,User-Agent
< Content-Length: 0
< Connection: close
< Content-Type: text/html; charset=UTF-8
```

Методи

1. ClientLogin

За да може да се използват методите от приложно-програмният интерфейс е нужно потребителите да се оторизират. Това трябва да става по следният начин. Прави се HTTP POST заявка към

<https://system/api/0/accounts/ClientLogin>. Подават се следните два параметъра:

- *Email*
- *Passwd*

Като отговор трябва да се получи токен, който трябва да се съхранява в клиента и да се подава при всяка следваща заявка. Например:

```
> POST /accounts/ClientLogin HTTP/1.1
> User-Agent: curl/7.19.7 (x86_64-redhat-linux-gnu) libcurl/7.19.7
NSS/3.14.0.0 zlib/1.2.3 libidn/1.18 libssh2/1.4.2
> Host: www.inoreader.com
> Accept: */*
> Content-Length: 27
> Content-Type: application/x-www-form-urlencoded
>
< HTTP/1.1 200 OK
< Date: Wed, 03 Feb2017 10:21:42 GMT
< Server: Apache
< X-Powered-By: PHP/7.1
< X-Mod-Pagespeed: 1.5.27.2-2912
< Vary: Accept-Encoding,User-Agent
```

```
< Cache-Control: max-age=0, no-cache
< Content-Length: 269
< Connection: close
< Content-Type: text/html; charset=UTF-8
<
Auth=xxxxxxxxxxxx
```

Полученият параметър *Auth* се съхранява в клиента и се използва при всяка следваща заявка като хедър:

```
Authorization: Token auth=xxxxxxxxxxxx
```

2. User information

Трябва да връща информация за текущо логнатият потребител. Прави се HTTP GET заявка към <https://system/api/0/user-info>. Трябва да се получава JSON обект със следната структура:

```
{
  userId: "1",
  userName: "customer",
  userEmail: " customer@innologica.com",
  userPicture: "http://addressofpicture.com/picture.jpg",
  signupTime: 1163850013,
}
```

Целта е приложенията да показват информация за текущо логнатият потребител

3. Списък с ключови думи

Трябва да връща списък с дефинирани ключови думи. Прави се HTTP GET заявка към <https://system/api/0/keywords>. Трябва да се

```
{
  keywords: [
    {
      id: "3614359203",
      keyword: "Президент",
    }
  ]
}
```

получава JSON обект със следната структура:

Трябва да бъде предоставен метод за странициране на резултатите, тъй като те може да са хиляди.

4. Списък със статии

Трябва да връща списък със статии спрямо ключова дума. Прави

се HTTP GET заявка към <https://system/api/0/articles/{keyword}> където {keyword} е url-encoded параметър – ключовата дума. Ако този параметър се проспусне трябва да се върнат всички статии. Трябва да се получава JSON обект със следната структура:

```
{
  direction: "ltr",
  updated: 1424637593,
  updatedUsec: "1424637593264558",
  items: [{
    crawlTimeMsec: "1422046342882",
    timestampUsec: "1422046342881684",
    id: "123",
    keywords: [
      "Политика",
      "БСП"
    ],
    title: "БСП и Подкрепа се харесаха, имали общи позиции",
    published: 1422046320,
    updated: 1422669611,
    canonical: [{
      href:
"http://www.dnes.bg/index/2017/02/24/bsp-i-podkrepa-se-haresaha-imali-obshti-pozicii.333015"
    }],
    alternate: [{
      href:
"http://www.dnes.bg/index/2017/02/24/bsp-i-podkrepa-se-haresaha-imali-obshti-pozicii.333015",
      type: "text/html"
    }],
    summary: {
      direction: "ltr",
      content: "..."
    },
    author: "Dnes",
    comments: [],
    origin: {
      streamId: "feed/http://dnes.bg",
      title: "Dnes.bg",
      htmlUrl: "http://www.dnes.bg/"
    }
  }
}]
}
```

Трябва да бъде предоставен метод за странициране на резултатите, тъй като те може да са хиляди.

Модул “Автоматично уведомяване”

- Изпращане на уведомления по email;
- Изпращане на push notifications към iOS устройства чрез използване на APNS (Apple Push Notifications Service) протокол;
- Изпращане на push notifications към Android устройства чрез използване на GCM (Google Cloud Messaging) протокол;
- Изпращане на SMS чрез добавяне на съвместим SMS

Gateway;

Модул “Автоматично уведомяване” трябва да реагира на заложените критерии при ключовите думи според които дадени съвпадение трябва да се изпращат към потребителите. Целта е дадени статии с особена важност да стигат до потребителите в реално време след постъване на статия в системата. Това трябва да става посредством опашка, която асинхронно да се разпределя към съответните канали:

- *Email*
- *SMS*
- *Push notifications (Android и iOS)*

Изпращането трябва да се осъществява непрекъснато като при грешка трябва да се правят последователни повторни опити с изчакване (т.нар. Exponential backoff).

Push нотификациите трябва да използват следните протоколи спрямо операционната система:

- *Android – Firebase Cloud Messaging (FCM) - <https://firebase.google.com/docs/cloud-messaging/>*
- *iOS – Apple Push Notification Service (APNS) - <https://developer.apple.com/notifications/>*
- *Windows Phone – Windows Push Notification Service (WNS) - <https://docs.microsoft.com/en-us/windows/uwp/controls-and-patterns/tiles-and-notifications-windows-push-notification-services--wns--overview>*

База Данни:

- **MySQL сървър версия 5.5;**
- **Използване на master – slave асинхронна репликация;**
- **Бекъпи на ключови таблици да се извършват на всеки час;**
- **Бекъпи на средни по важност таблици да се извършват един път дневно;**
- **Бекъпи на съдържание на статии да се извършва един път седмично. Повечето медийни източници поддържат отдалечено статии и могат да бъдат възстановени;**

Поради очакваните огромни обеми от данни, базите трябва да оптимизирани и готови да поемат натоварването, като при нужда решението трябва да може да се скалира хоризонтално чрез добавяне на нови сървъри и без необходимост от преработки.

Релационни бази (RDBMS): *Системата трябва да използва MySQL версия 5.5 или по-нова. Позволява се и използването на Persona Server. Задължително таблиците да бъдат с InnoDB/XtraDB engine. Архитектурата на базата трябва да е така организирана, че да е възможно хоризонтално скалиране, тъй като се очаква в системата постепенно да се натрупат огромно количество данни, които не биха се побрали на един физически сървър.*

<p>Документални бази (NoSQL): За нуждите на търсене трябва да се използва <i>Elasticsearch</i>, като най-модерно решение. Задължително е индекса да бъде коректно настроен с поддръжка на най-популярните езици, включително CJK. Поддръжката не става автоматично, а трябва мапингите прецизно да се настроят за всеки език.</p> <p>Файлов сървър:</p> <ul style="list-style-type: none"> – Да поддържа статични HTML и PDF файлове с генерирани репорти с архивна цел; – Файловете да са достъпни чрез HTTP интерфейс и токен оторизация; <p>Мобилни приложения:</p> <ul style="list-style-type: none"> – Разработване на мобилно приложение за Android, написано изцяло на Java с помощта на Android Studio; – Разработване на мобилно приложение за iOS, написано на Objective C с помощта на Xcode; – Разработване на мобилно приложение за Windows Phone, написано изцяло на C# с помощта на Visual Studio; <p><i>Мобилните приложения трябва да боравят с приложно-програмният интерфейс и да визуализират информация на потребителите, да дават възможност за дефиниране и промяна на ключови думи, правила и действия; да е възможно разглеждането на репорти, както и получаването на Push нотификации от модулът за автоматично уведомяване. Основният език на приложенията трябва да бъде Английски, като трябва да има опция за смяна на езика. Трябва да се поддържа и Български, както и да бъде предложена схема за превеждане от потребители.</i></p> <p><u>Задължителни екрани в приложенията:</u></p> <ul style="list-style-type: none"> – <i>Login и регистрационни екрани</i> <i>Тези екрани трябва да се поддържат основните методи за регистрация и вписване, които са Email/парола или социални мрежи (Facebook, Google, LinkedIn). Тези методи трябва да бъдат разработени в приложението и бекенда.</i> <ul style="list-style-type: none"> • <i>Facebook login</i> <ul style="list-style-type: none"> ○ <i>iOS -</i> https://developers.facebook.com/docs/facebook-login/ios ○ <i>Android -</i> https://developers.facebook.com/docs/facebook-login/android • <i>Google login</i> 		
--	--	--

<ul style="list-style-type: none"> ○ iOS - https://developers.google.com/identity/sign-in/ios/start-integrating ○ Android - https://developers.google.com/identity/sign-in/android/start-integrating ● LinkedIn login <ul style="list-style-type: none"> ○ iOS - https://developer.linkedin.com/docs/ios-sdk-auth ○ Android - https://developer.linkedin.com/docs/android-sdk-auth <p>– Екрани за манипулация на ключови думи Тези екрани трябва да се извежда списък със всички дефинирани от потребителя ключови думи. При тапване върху ключова дума трябва да се влиза в резултатите за нея (статии). При задържане на пръста върху която и да е ключова дума трябва да се показва контекстно меню с 4 избора:</p> <ul style="list-style-type: none"> ● Преглед на резултатите – отваря списък със статии ● Репорти – отваря всички репорти в които участва тази ключова дума ● Редактиране – дава възможност за промяна на ключовата дума ● Изтриване – изтрива ключовата дума. Трябва да се мине през потвърждение на потребителя <p>– Екран за настройки</p> <p>– Екрани за статии Списъците със статии трябва да съдържат заглавие на статията, дата, източник и кратко резюме на съдържанието. При тапване върху статия трябва да се влиза в цялостният ѝ изглед както е бил в RSS summary полето или content от JSON интерфейс. При тапване на Share бутона горе в дясно трябва да се отваря стандартният sharing sheet на iOS или sharing intent на Android. При тапване върху заглавието на статията, тя следва да се отвори в браузър, като той се навигира към оригиналното URL на статията.</p> <p>– Главно меню “хамбургер” Главното меню трябва да служи за бърза навигация от която и да е точка на приложението. В него трябва да има следните елементи:</p> <ul style="list-style-type: none"> ● Настройки ● Статии 		
---	--	--

- Източници
- Ключови думи
- Репорти
- Изход (отписване от акаунта, не реален изход)

Към техническите и функционални характеристики, които предлага Кандидатът, трябва да бъде представена демонстрационна визуализация – следните макети (визуализации) на екрани в статичен или динамичен вид (в pdf/jpeg/png/html или еквивалентен формат) за Специализирана софтуерна платформа за медиен мониторинг, с графично и текстово съдържание, съгласно посоченото:

Макети на екрани (web) за Модул “Потребителски” в статичен или динамичен вид (в pdf/jpeg/png/html или еквивалентен формат)

	Макет на екран/и	Описание
1	Презентационен портал	Макетът трябва да изобразява формата за логин чрез потребителско име и парола, линк за възстановяване на забравена парола, както и трите алтернативни метода за логин - Google, Facebook, LinkedIn. Също така трябва да е видима опцията за промяна на езика.
2	Настройки	Макетът трябва да показва панел с настройки. Следните настройки трябва да бъдат показани: <ul style="list-style-type: none"> • Име • Потребителско име • Email • Парола • Език • Честота на опресняване на новините • Включване/изключване на известия за новини (чрез Web Notifications)
3	Репорти	Макетът трябва да показва отворен репорт. Репорта представлява списък със статии от съвпаденията по ключови думи, както и следната информация:

		<ul style="list-style-type: none"> • Графика на съвпаденията по дни • Гъстота на ключовите думи (коя дума се среща най-често в репорта) • Езици на статиите 		
4	Ключови думи и правила	<p>Трябва да бъдат представени <u>два</u> екрана - един със списък от ключовите думи и втори - с конфигурация на определена дума.</p> <p>Първият екран трябва да показва списък с конфигурираните потребителят ключови думи. Трябва да има бутон за добавяне на нова ключова дума. В списъка трябва да се вижда думата, бърз бутон за активация/деактивация, бутон за изтриване както и брой попадения за днешният ден.</p> <p>Вторият екран е конфигурация на дадена ключова дума. Трябва да се отваря в дясната част на екрана като модален диалог. Трябва да може да се конфигурира името, както и списък с правила спрямо изискванията, посочени в секция "Ключови думи и правила" от Техническата спецификация към процедурата, както и да има бутони за запис (Save) и отказ (Cancel).</p>		
5	Статии	<p>Трябва да бъдат представени <u>четири</u> екрана - за режим "Списък", режим "Списание", режим "Картички", както и екран с отворена статия.</p> <p>Над списъка със статии трябва да има ясен компонент (бутони) за промяна на изгледа. В режим "Списък" трябва всяка статия да заема един ред от</p>		

		<p>екрана като е видим източникът на статията, нейното заглавие и датата ѝ. Източниците трябва да бъдат обозначени със икони както и име.</p> <p>В режим "Списание" статиите трябва да бъдат представени като контейнери с фиксирана ширина и височина. В лявата част на контейнера трябва има снимка от статията, а в дясната трябва да са видими заглавието, датата, и източникът.</p> <p>Височината на статиите трябва да бъде поне 80 пиксела, широчината поне 600 пиксела.</p> <p>В режим "Картички" статиите са представени като списък от контейнери с фокус върху изображението, което трябва да заема над 90% от размера на целият контейнер. Заглавието на статията и датата трябва да бъдат разположени върху самото изображение с подходяща маска или сянка на текста така че да бъде четим при всякакви условия.</p> <p>Източникът на статията може да бъде под изображението, но отново да бъде част от картичката. Препоръчителни размери: хоризонтално 300 пиксела, вертикално 240 пиксела.</p> <p>Екранът с отворена статия трябва да показва модален диалог отворен в дясната част на екрана. Най-отгоре трябва да бъде видим източника на статията, след това заглавието с подходящ голям шрифт. Под него датата и след това съдържанието на самата статия.</p>		
--	--	--	--	--

6	Табло	<p>Макетът трябва да показва:</p> <ul style="list-style-type: none"> • Новини от системата (обновления) • Линкове към мобилните устройства • Брой нови статии от последното влизане • Генерирани нови репорти • Списък с ключовите думи и брояч след всяка дума показващ броя съвпадения за дена • Непрочетени нотификации • Графика на статиите в репорти по дни • Графика на гъстотата на ключовите думи 		
7	Административен портал	<p>Трябва да бъдат представени три екрана - Емисии, Потребители и Плащания.</p> <p>Екранът с емисии трябва да показва таблица с емисиите в системата. Трябва да са видими името, рейтинг на емисията, статии които се публикуват на ден, последно обновяване, статус от последното обновяване.</p> <p>Екранът с потребители трябва да показва таблица със следните полета: Име на потребителя, ел. поща, Държава, Дата на регистрация, Реферер (от къде е регистриран), бутон за изтриване. Също така трябва да има графика с броят потребители, които са онлайн в момента в системата, както и статистика за новите регистрации за последният час, ден, седмица и месец.</p> <p>Екранът с плащания трябва да показва таблица със следните полета: Потребител, Ел. поща, Дата, Сума, План, Статус.</p>		
<p>Макети на екрани за Мобилни приложения <u>в статичен или динамичен вид</u> (в pdf/jpeg/png/html или еквивалентен формат)</p>				

	Макет на екран/и	Описание		
1	Логин екран	Макетът трябва да изобразява формата за логин чрез потребителско име и парола, както и трите алтернативни метода за логин - Google, Facebook, LinkedIn. Език – Български или Английски.		
2	Регистрационен екран	Макетът трябва да изобразява формата за регистрация чрез потребителско име и парола, както и трите алтернативни метода за регистрация - Google, Facebook, LinkedIn. Език – Български или Английски.		
3	Забравена парола	Макетът трябва да изобразява формата за изпращане на забравена парола по мейла. Език – Български или Английски.		
4	Ключови думи	Макетът трябва да представлява Списък от свързани екрани, които трябва да изобразяват интеракции с ключови думи – Разглеждане на списъка с ключови думи, Изтриване, Редактиране, Разглеждане на репортите в които ключовата дума участва, Разглеждане на статиите записани към всяка ключова дума. Език – Български или Английски.		
5	Настройки	Макетът трябва да изобразява екран със следните настройки: Име, Email, Парола, Език Език – Български или Английски.		
6	Статии	Макетът трябва да представлява Списък от свързани екрани, които трябва да изобразяват интеракции със статии – Списък със статии -> Единична статия -> Шерване на статията към външни приложения, Преглед на статията в браузър. Език – Български или Английски.		
7	Главно меню	Макетът трябва да изобразява Главно меню – тип "Хамбургер меню" с бутон горе в ляво, чрез който да се отваря и затваря. В менюто трябва да присъстват следните елементи - Настройки, Статии, Източници, Ключови думи, Репорти, Изход. Език – Български или Английски.		

<p>Екраните за Android, iOS и Windows Phone трябва да бъдат еднакви. Допускат се Wireframe прототипи.</p>		
<p>Изисквания към гаранционната и извънгаранционната поддръжка (ако е приложимо):</p> <p>1. Гаранционен срок – (да се посочи в месеци) Максималният гаранционен срок, който участниците в процедурата могат да оферират е 36 месеца;</p> <p>2. Време за отстраняване на повреда – (да се посочи в часове) Минималното време за отстраняване на повреда, което участниците в процедурата могат да оферират е 1 час.</p>		
<p>Изисквания към документацията, съпровождаща изпълнението на предмета на процедурата (ако е приложимо): Активът трябва да бъде доставен с техническа документация на български език или, ако няма такава възможност, на английски език. Електронни версии са приемливи вместо хартиен носител.</p>		
<p>Изисквания към правата на собственост и правата на ползване на интелектуални продукти (ако е приложимо). Прехвърляне на всички авторски права и интелектуална собственост върху <u>Специализирана софтуерна платформа за медиен мониторинг</u> от Изпълнителя към Възложителя. Авторското право върху актива, предмет на договорът за изработка, който предстои да се сключи, възниква от момента на неговото създаване изключително за ВЪЗЛОЖИТЕЛЯ, по чиято поръчка и за чиято сметка и риск е създаден. Авторските права, доколкото възникват такива, върху софтуерния продукт и неговите елементи, сорс код на софтуер; бази данни със собствена или събирана информация; задание; чертеж; план;</p>		

модел; прототип; изделие; технология или процес; научен, промишлен или търговски опит (know how); научен или пазарен анализ, създадени, при или по повод изпълнение на задълженията по чл. 1 на договора за изработка, който предстои да се сключи, възникват и принадлежат изключително на ВЪЗЛОЖИТЕЛЯ от момента на създаването им.		
Изисквания за обучение на персонала на бенефициента за експлоатация : Няма		
Подпомагащи дейности и условия от бенефициента (ако е приложимо). Няма		
Други: Няма		

При така предложените от нас условия, в нашето ценово предложение сме включили всички разходи, свързани с качествено изпълнение на предмета на процедурата в описания вид и обхват, както следва:

ЦЕНОВО ПРЕДЛОЖЕНИЕ

I. ЦЕНА И УСЛОВИЯ НА ДОСТАВКА

Изпълнението на предмета на процедурата ще извършим при следните цени:

№	Описание на доставките/услугите/дейностите/ строителството	К-во /бр./	Единична цена в лева (с изключение на процедурите с предмет услуги)	Обща цена в лева без ДДС (не се попълва при извършване на периодични доставки)
1				
2				
3				
4				

За изпълнение предмета на процедурата в съответствие с условията на настоящата процедура, общата цена¹ на нашата оферта възлиза на:

Цифром: _____ Словом: _____
(посочва се цифром и словом стойността без ДДС)

Декларираме, че в предложената цена е спазено изискването за минимална цена на труда (за случаите, когато процедурата е за избор на изпълнител на договор за строителство).

¹ Не се посочва при извършване на периодични доставки.

II. НАЧИН НА ПЛАЩАНЕ

Предлаганият от нас начин на плащане е, както следва: _____
(описва се)

При разминаване между предложените единична и обща цена, валидна ще бъде общата цена на офертата. В случай че бъде открито такова несъответствие, ще бъдем задължени да приведем единичната цена в съответствие с общата цена на офертата.

При несъответствие между сумата, написана с цифри, и тази, написана с думи, важи сумата, написана с думи.

Като неразделна част от настоящата Оферта, прилагаме следните документи:

1. Декларация с посочване на ЕИК/Удостоверение за актуално състояние;
 2. Декларация по чл. 12, ал. 1, т. 1 .от Постановление № 160 на Министерския съвет от 2016 г.;
 3. Доказателства за икономическо и финансово състояние (ако такива се изискват);
- **Отчет за приходите и разходите (или еквивалент) като съставна част от годишния финансов отчет за последните 3 приключени финансови години, в зависимост от датата, на която кандидатът е учреден или е започнал дейността си – копие заверено от кандидата с подпис, печат и текст „Вярно с оригинала“.**
 4. Доказателства за технически възможности и/или квалификация (ако такива се изискват);
- **Документ, удостоверяващ въведена в предприятието на кандидата сертифицирана Система за управление на качеството по стандарт ISO 9001 или еквивалент - копие на валиден сертификат, заверено от кандидата с подпис, печат и текст „Вярно с оригинала“.**
 - **Документ, удостоверяващ въведена в предприятието на кандидата сертифицирана Система за управление на сигурността на информацията по стандарт ISO 27001 или еквивалент - копие на валиден сертификат, заверено от кандидата с подпис, печат и текст „Вярно с оригинала“.**
 - **Документ, удостоверяващ въведена в предприятието на кандидата сертифицирана Система за управление на качеството на ИТ услугите по стандарт ISO 20000-1 или еквивалент - копие на валиден сертификат, заверено от кандидата с подпис, печат и текст „Вярно с оригинала“.**
 - **Документи, удостоверяващи, че ключовият експерт – Ръководител на разработката, притежава сертификат за придобита квалификация. Представя се копие на валиден Сертификат Agile Software Development with SCRUM или еквивалент, заверено от кандидата с подпис, печат и текст „Вярно с оригинала“.**

5. Декларация за подизпълнителите, които ще участват в изпълнението на предмета на процедурата и дела на тяхното участие (*ако кандидатът е декларирал, че ще ползва подизпълнители*);

6. Документи по т. 1, 2 ,3 и 4 за всеки от подизпълнителите в съответствие с Постановление № 160 на Министерския съвет от 2016 г. (*когато се предвижда участието на подизпълнители*);

7. Други документи и доказателства, изискани и посочени от бенефициента в документацията за участие – **в случай, че кандидатът е обединение на физически и/или юридически лица, следва да се представи Договор за обединение или документ, подписан от лицата в обединението, в който задължително се посочва представляващият, както и да се предоставят документите по т. 1 и 2 за всяко лице от обединението.**

ДАТА: _____ г.

ПОДПИС и ПЕЧАТ: _____

(име и фамилия)

(длъжност на представляващия кандидата)